

Coral reefs restoration using Machine Learning

N. Sai Karthik

Dept. of Computer Science and Engineering

SRM University

Amaravathi, Andhra Pradesh, India.

saikarthik_nallamotheu@srmap.edu.in

Abstract—Coral reefs, one of the most diverse and valuable ecosystems on our planet, are facing unprecedented threats due to climate change, pollution, and human activities. To combat the alarming decline of coral reefs, innovative and efficient restoration techniques are urgently needed. Machine learning algorithms have demonstrated remarkable capabilities in analyzing complex ecological data and providing valuable insights for decision-making. In the context of coral reef restoration, ML can be employed to address key challenges such as efficient identification of suitable restoration sites, optimizing deployment strategies, and enhancing coral health monitoring. The usage of machine learning into coral reef restoration efforts holds great promise for the rehabilitation and conservation of these critically endangered ecosystems. ML techniques can provide valuable insights, enhance decision-making processes, and optimize restoration strategies, ultimately contributing to the long-term resilience and sustainability of coral reefs.

Keywords—deployment, monitoring, decision-making processes

1. Introduction

In response to the critical state of coral reefs, restoration efforts have gained considerable momentum in recent years. Traditional restoration methods, such as coral transplantation and the creation of artificial structures, have shown promise but often face

limitations in terms of efficiency, scalability, and long-term success. To overcome these challenges and accelerate the recovery of coral reefs, innovative approaches are urgently needed. One such promising avenue lies in the intersection of coral reef restoration and machine learning (ML). Machine learning is a branch of artificial intelligence that focuses on developing algorithms capable of learning and making predictions or decisions based on data. ML algorithms have demonstrated remarkable capabilities in various fields, including image recognition, natural language processing, and environmental modeling. When applied to coral reef restoration, ML has the potential to revolutionize the way we approach rehabilitation efforts and increase their effectiveness. As we delve into the various applications and implications of ML in coral reef restoration, we uncover new avenues for research, collaboration, and innovation in the quest to safeguard these invaluable ecosystems for future generations.

2. Problem Description

2.1 Case Study

Coral reefs are some of the most diverse and important ecosystems in the world, both for marine life and society more broadly. Not only are healthy reefs critical to fisheries and food security, they also protect coastlines from storm surge, support tourism-based economies, and advance drug discovery research, among other countless benefits. Reefs face a number of rising threats, most notably climate change, pollution, and overfishing. In the past 30 years alone, there

have been dramatic losses in coral cover and habitat in the Great Barrier Reef (GBR), with other reefs experiencing similar declines. The outbreaks of the coral-eating crown of thorns starfish (COTS) have been shown to cause major coral loss. While COTS naturally exist in the Indo-Pacific, reductions in the abundance of natural predators and excess run-off nutrients have led to massive outbreaks that are devastating already vulnerable coral communities. Controlling COTS populations is critical to promoting coral growth and resilience.

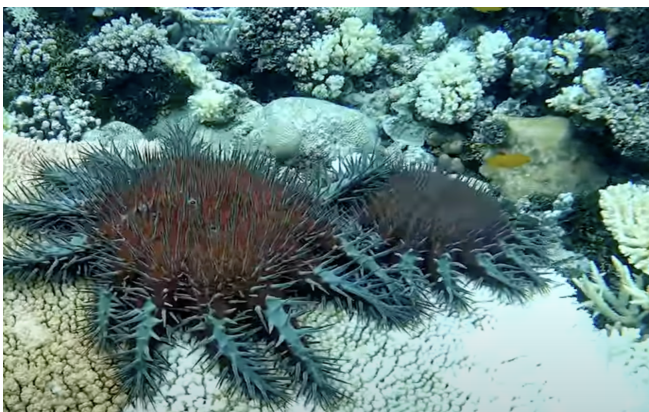


Figure 1. Coral eating starfish

2.2 Approach

First we need to do the data analysis. Data analysis is the process of collecting, cleaning, transforming, and modeling data with the target of discovering and extracting useful information, making conclusions, and decision-making. Data pre-processing is used to remove the number of discrepancies associated with the data, remove duplicate records, normalize values, account for missing data, etc. The primary step in this data pre-processing is to check for null values and treat them by filling in or dropping them. After importing a dataset using the Python library pandas, common data pre-processing methods such as data cleaning, data transformation, efficient processing, and classification are performed. No unique method of data processing is used in this work.

The data set must be splitted into test data and training data So, that we can compare actual and predicted outputs. Yolo V5 and TensorFlow are used in this research. Tensor Flow is an open source machine learning library which is used for training and inference of deep neural networks whereas Yolo V5 is also an open source library in pytorch which is used for object detection and image classification.

Artificial Neural Networks (ANN) are a type of machine learning model which are similar to neurons in the brain. It teaches machine learning to learn similar to human brain

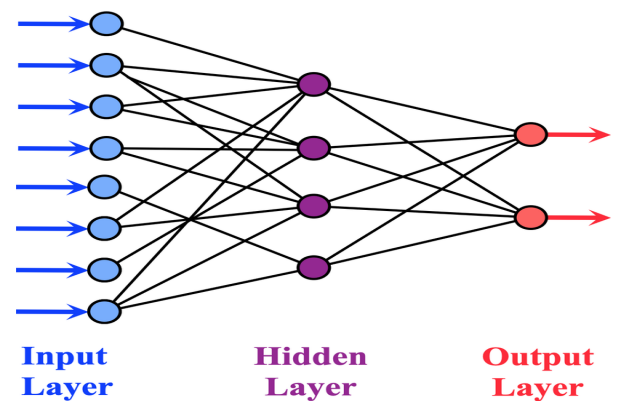


Figure 2. Simple Neural Network

The dataset should be trained and then later inference must be applied on the model to evaluate the output. It took around 8 hours for the model to get trained.

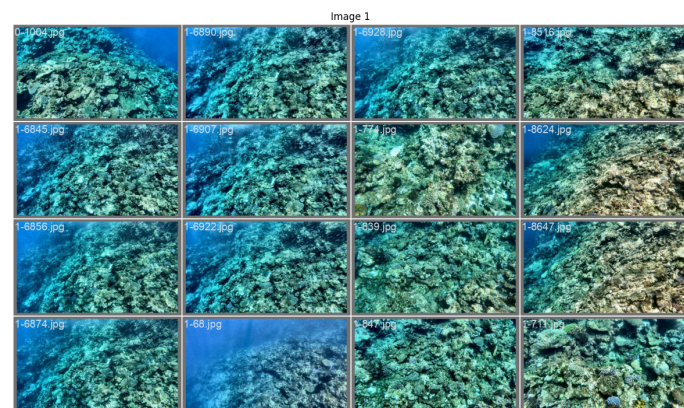


Figure 3. Sample test data

During the training, Meta data for training images were loaded into pandas dataframe and then percentage of images with and without bounding boxes, which is crucial for model training. Images are copied to the working directory since the original input directory does not have write access, which is necessary for YOLOv5 training.

YOLOv5 utilizes a convolutional neural network (CNN) architecture to process images. It consists of a backbone network, neck, and head, with multiple convolutional layers designed to detect objects.

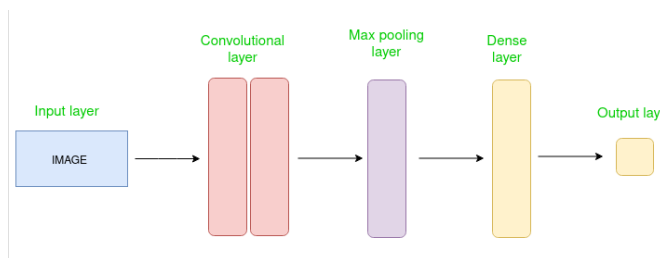


Figure 4. CNN in deep learning

Utility functions are defined for working with bounding boxes, image sizes, and labels. Bounding box data is extracted and formatted from the annotations in the dataset. The dimensions of all images in the dataset are determined and recorded. Bounding box data is converted from COCO format to YOLO format, and label files are created for YOLO model training. The data is split into different folds using GroupKFold for cross-validation.

2.3. Experimental Setup

Pictures and videos of corals were taken with the help of underwater cameras.



Figure 5. Capturing real time footage

Then the raw footage and values were made into a data set in a folder containing training set photos, video id and video jpeg format. Below are the contents of the data set.

- [train/test].csv - Metadata for the images. As with other test files, most of the test metadata data is only available to your notebook upon submission. Just the first few rows available for download.
- video_id - ID number of the video the image was part of. The video ids are not meaningfully ordered.
- video_frame - The frame number of the image within the video. Expect to see occasional gaps in the frame number from when the diver surfaced.
- sequence - ID of a gap-free subset of a given video. The sequence ids are not meaningfully ordered.
- sequence_frame - The frame number within a given sequence.
- image_id - ID code for the image, in the format {video_id}-{video_frame}
- Annotations - The bounding boxes of any starfish detections in a string format that can be evaluated directly with Python. A bounding box is described by the pixel coordinate (x_min, y_min) of its lower left corner within the image together with its width and height in pixels --> (COCO format).

Each prediction row needs to include all bounding boxes for the image [x_min, y_min, width, height]. Competition metric F2 tolerates some false positives (FP) in order to ensure very few starfish are missed. Which means tackling false negatives (FN) is more important than false positives (FP).

$$F2 = 5 \cdot \alpha / (4 + \alpha)$$

Where, α = precision/recall

Along with them Weights & Biases were used in the model. It is a MLOps platform for tracking experiments. We can use it to Build better models faster with experiment tracking, dataset versioning, and model management. We compare, and visualize ML experiments, get live metrics, terminal logs, and system stats streamed to the centralized dashboard. Explain how your model works, show graphs of how model versions improved, discuss bugs, and demonstrate progress towards milestones.

3. Results

Imbalance classification is a problem in machine learning where classes may be biased or skewed. The distribution can vary from a slight bias to a severe imbalance where there is one example in the minority class for hundreds, thousands, or millions of examples in the majority class or classes.

In Figure 6, We can see the batch images of the corals with starfishes generated in the output

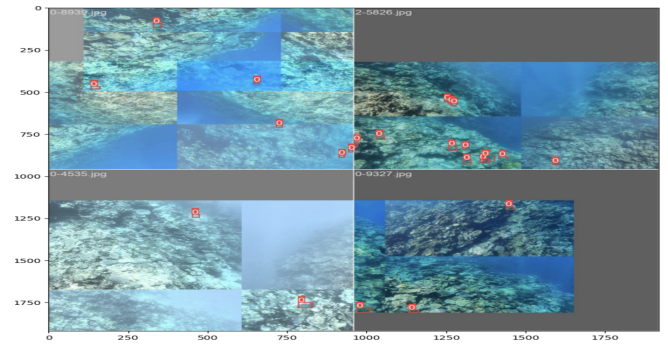


Figure 6. Batch Image of COTS.

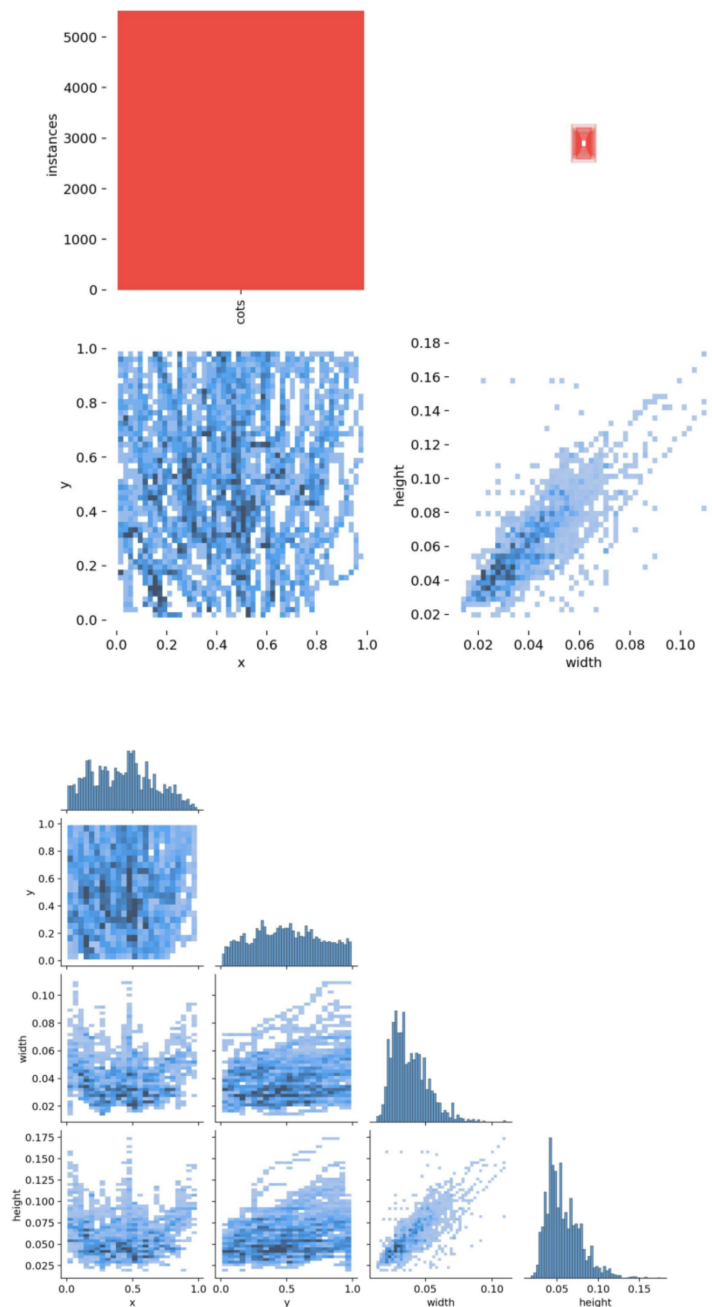


Figure 7. Class Distribution Images

Confusion matrix was used to show the predicted and background of the COTS. Figure 8 shows the confusion of matrix of the model

These are some examples of the confusion matrix:

True Positive: The model's prediction was matched by the actual value, which was yes.

True Negative: Both the actual and projected values were negative.

False Positive: The actual value was Yes and model prediction was No this is called Type1 error.

False Negative: The actual value was No and model prediction was Yes this is called Type2 error.

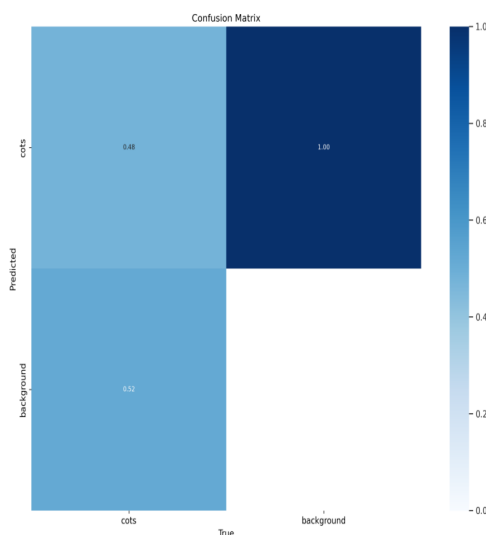


Figure 8. Confusion Matrix

3.1 Evaluation Metrics

After the modeling process, the main task is to evaluate the model on the test data to check whether a particular machine learning algorithm to predict correctly.

AccuracyScore: The proportion of accurate predictions to all input data points is what determines the classification accuracy score. Accuracy score is equal to the proportion of correct predictions to all data points.

$$TP + TN Accuracy = TP + TN + FN + FP$$

Precision: The precision ratio is the sum of the predicted positives divided by the number of true positives.

$$Precision = TP / TP + FP$$

Recall: The proportion of True Positives to all other Positives is known as the recall rate.

$$TRP/Recall/Sensitivity = TP / TP + FN$$

F1_score: Precision and recall together make up the F1 score. Only when recall and precision are high is the F1 score high.

$$F1 score = 2 * Precision * Recall$$

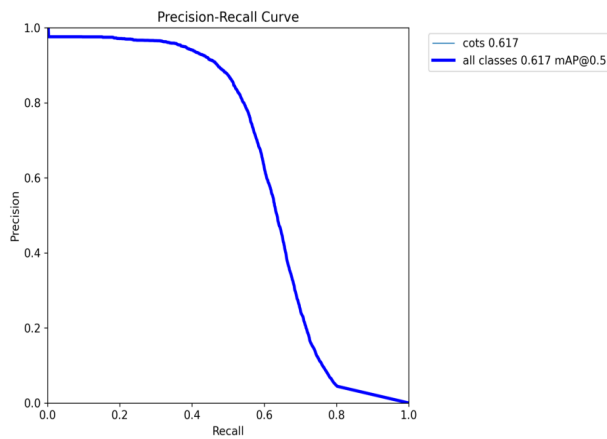


Figure 9. Precision Recall Curve

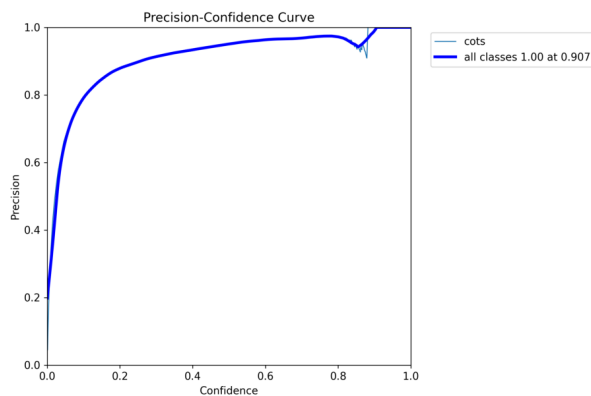


Figure 10. Precision Confidence Curve

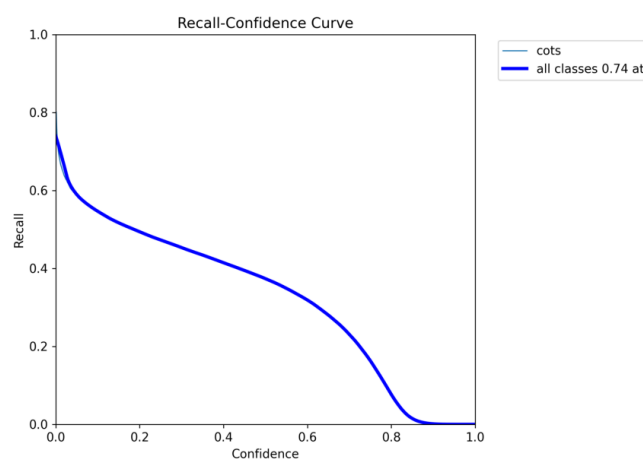


Figure 11. Recall-Confidence Curve

4. Conclusion

Our research presents a robust and practical solution for the detection of crown-of-thorns starfish in underwater images, contributing to the conservation efforts of coral reefs. This research addresses the main issue of corals being damaged by COTS but not only starfish but many corals have declined due to the ocean pollution. We addressed only a small part, But in future this research continues to expand and explores all possible restoration of corals using AI and ML models.

5. References

- [1] *Using Machine Learning to Help Protect the Great Barrier Reef in Partnership with Australia's CSIRO.* (n.d.). <https://blog.tensorflow.org/2022/05/Kaggle-Great-Barrier-Reef-ML.html>
- [2] *TensorFlow - Help protect the great barrier reef Kaggle.* (n.d.). <https://www.kaggle.com/c/tensorflow-great-barrier-reef/>
- [3] Schürholz, D., & Chennu, A. (2022). Digitizing the coral reef: Machine learning of underwater spectral images enables dense taxonomic mapping of benthic habitats. *Methods in Ecology and Evolution*, 14(2), 596–613. <https://doi.org/10.1111/2041-210x.14029>
- [4] P. L. Jokiel and J. Naughton, "Coral reef mitigation and restoration techniques employed in the Pacific Islands: I. Overview," Oceans 2001 MTS/IEEE Proceedings, Holland Publications, Escondido, California (this volume).